Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Rok Mirt

Študija varnosti OBD Bluetooth adapterjev

diplomsko delo

univerzitetni študijski program
prve stopnje
računalništvo in informatika

Mentor: izr. prof. dr. Patricio Bulić

Ljubljana, 2017

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Na tržišču je veliko cenenih adapterjev za avtomobilsko diagnostiko (On-Board Diagnostics). V diplomskem delu preučite uporabi in varnost cenenih OBD Bluetooth adapterjev. Preučite možne napade na centralni računalnik s pomočjo cenenih OBD adapterjev. Izdelajte aplikacijo na sistemu Android s katero preizkusite delovanje dveh OBD adapterjev.

Za pomoč pri izdelavi diplomskega dela se zahvaljujem Mateju Peršolji in Metodu Ribiču, ki sta s tehničnimi nasveti pomagala pri raziskavi, ter mentorju izr. prof. dr. Patriciu Buliću. Rad bi se zahvalil tudi svoji družini, ki me je v času študija pri tem podpirala.

Kazalo

Seznam uporabljenih kratic

kratica	angleško	slovensko
OBD	On Board Diagnostic	Avtomobilska diagnostika
ECU	Engine Control Unit	kontrolna enota motorja
EU	European Union	Evropska Unija
USB	Universal Serial Bus	Univerzalno serijsko vodilo
RPM	Revolutions Per Minute	Število obratov na minuto
SAE	Society of Automotive Engineers	(Združenje inženirjev, ki se ukvarjajo z vozili vseh vrst.)
CRC	Cyclic Redundancy Check	Ciklična redundančna koda
ISO	International Standards Organisation	(mednarodna organizacija za standarde)
PID	Parameter ID	ID parametra

Povzetek

Naslov: Študija varnosti OBD Bluetooth adapterjev

Avtor: Rok Mirt

Dandanes imajo že skoraj vsi avtomobili vmesnik OBD (On Board Diagnostics). Ta se običajno uporablja za odkrivanje napak v vozilu.V tem diplomskem delu smo raziskali možnosti uporabe Bluetooth adapterja za komunikacijo z OBD ter varnost samega OBD. S pomočjo mobilne aplikacije smo poskušali prebrati standardne OBD podatke, kot so recimo hitrost vozila, število obratov motorja, itd. Raziskali smo tudi kako je s stabilnostjo takega adapterja in tudi samega OBD.
Z adapterjem ni bilo večjih težav, razen občasnih izpadov povezave. Preko OBD nam je uspelo pridobiti podatke, ki smo jih omenili zgoraj. O OBD pa je odkritih že precej ranljivosti, nekatere so zelo resne, na primer možnost oddaljenega pospeševanja, ali izklop zavor.

Ključne besede: OBD, Bluetooth, Android, varnost, vozilo.

Abstract

Title: Safety of the OBD Bluetooth adapters: a case study

Author: Rok Mirt

Nowadays almost every vehicle has OBD (On Board Diagnostics) interface. It is most commonly used for car maintenance. We researched Bluetooth adapter usage (read data like RPM or vehicle speed) and security of OBD itself. We tried to read standard OBD data (vehicle speed, RPM...) using mobile app. We researched the adapter stability, as well as OBD's.
There were some minor issues with adapter (dropping connection). We managed to get the data we mentioned before. On the contrary OBD is known to have several (possibly deadly) vulnerabilities, for example someone being able to accelerate or apply brakes remotely.

Keywords: OBD, Bluetooth, Android, security, vehicle.

Poglavje 1

Uvod

Vsako vozilo, ki je bilo narejeno po letu 1996 (v ZDA) [1] , ali 2001 (z bencinskim motorjem) oziroma 2004 (z dizelskim motorjem) v EU [2], mora imeti po zakonu tudi standardni vhod za dostop do OBD-II. Ta se običajno uporablja na servisu za diagnostiko avtomobila. Do podatkov pa lahko dostopamo tudi sami in sicer preko cenovno ugodnih USB (Universal Serial Bus) ali Bluetooth adapterjev.

Tako lahko iz kontrolne enote motorja (angl. ECU - Engine Control Unit, tudi Electronic Control Unit) pridobimo podatke, kot so recimo hitrost vozila, število obratov na minuto (angl. RPM - Revolutions Per Minute), nivo goriva in druge. Včasih nas ti podatki med vožnjo zanimajo. Če smo nekoliko bolj zahtevni, lahko uporabimo aplikacije, ki nam recimo izračunajo porabo goriva, ali pa ocenijo našo vožnjo. Zavarovalnice že ponujajo avtomobilsko zavarovanje po meri [3], kar dosežejo z uporabo namenskih aplikacij, ki pa na žalost niso tako natančne [4]. Tisti še bolj zahtevni, pa lahko napišejo aplikacijo po svoji meri.

Pri vsem tem pa se pojavi vprašanje varnosti. Kako je s stabilnostjo vozila, če iz ECU beremo podatke med vožnjo? Na servisu se to dogaja v kontroliranem okolju, česar pa v praksi ne moremo pričakovati. Ali je možno „pokvariti“ vozilo z uporabo poceni adapterjev?

Vse to smo poskušali ugotoviti v tem diplomskem delu. V ta namen

smo napisali tudi enostavno Android aplikacijo, ki je služila kot prejemnik podatkov, ki jih je OBD adapter prebral. Aplikacija jih obdela in poskuša iz njih izluščiti kak uporabniku uporaben podatek, ter ga prikazati na zaslonu.

1.1 Cilji

Kot že omenjeno, je cilj te diplomske naloge raziskati, kako je z varnostjo OBD, poleg tega pa bi radi raziskali tudi, kaj vse je možno preko OBD prebrati. Če cilje razdelimo na več delov:

- Delujoča Android aplikacija
- Seznam vseh možnih prebranih podatkov
- Varnost OBD
- Poglobiti znanje o delovanju OBD

1.2 Pregled področja

Področje OBD-II je že kar dobro pokrito. Obstaja že precej aplikacij, ki znajo iz vmesnika OBD prebrati podatke (običajno s pomočjo Bluetooth ali USB / RS232 žičnih adapterjev). Mi bomo spisali svojo iz preprostega razloga - nadzor nad celotnim dogajanjem (za potrebe študije). Na spletu se najde nekaj zapisov o tem, da nam lahko ceneni adapterji (običajno narejeni na Kitajskem) uničijo ECU (o tem priča kar nekaj uporabnikov)[5].

Poglavje 2

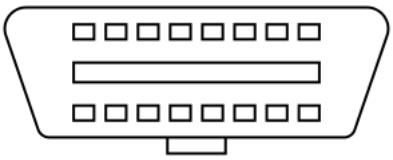
Raziskava

2.1 OBD-II

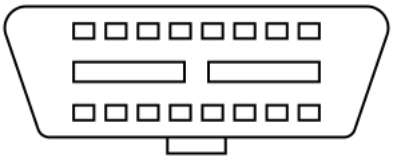
Sledi opis OBD-II, saj je ta zelo pomemben za nadaljno raziskavo.

Kaj je OBD? OBD (angl. OnBoard Diagnostics) se nanaša na zmožnost samodiagonstike vozila. Tako imenovani diagnostični računalnik (angl. on-board computer) se je začel pojavljati okrog leta 1969. Težava je bila, da je vsak proizvajalec vozil uporabljal svoj lasten (angl. proprietary) vmesnik za dostop do le tega. Tako je moral mehanik (oziroma kdorkoli je želel narediti diagnostiko) od proizvajalca vozil kupiti napravo, ki je znala komunicirati z diagnostičnim računalnikom, in sicer za vsako znamko posebej, včasih pa celo za različne modele iste znamke. Vse skupaj je bilo seveda drago, zato za navadne smrtnike to seveda ni prišlo v poštev.

Kasneje so se konektorji, vmesnik za dostop do diagnostičnega računalnika, vodilo in protokoli za komuniciranje z računalnikom standardizirali po SAE J1962 [6] oziroma ISO 15031-3:2004 [7] v kar danes poznamo kot OBD-II. S standardizacijo je nakup naprave za komuniciranje z računalnikom postal manjši strošek, z vstopom v informacijsko dobo pa je ta postal tako majhen, da danes lahko naprave, ki znajo komunicirati z diagnostičnim računalnikom, kupimo (v tujini) za nekaj ameriških dolarjev oziroma evrov.



Slika 2.1: OBD-II konektor tip A - ženski



Slika 2.2: OBD-II konektor tip B - ženski

Zakonska podlaga V Združenih državah Amerike je OBD-II zakonsko obvezen v vseh vozilih izdelanih od leta 1996 naprej [1]. Od junija 2000 naprej je v vseh vozilih obvezen v Evropi EOBD standard, ki je v osnovi enak OBD-II [2].

Konektorji Uporabljata se dva standardna (SAE J1962)[6] konektorja, tip A in tip B. Večjih razlik med njima ni, le da je tip B sestavljen iz dveh lukenj na sredini, tip A pa ima samo eno čez celotno površino (glej sliki 2.1 in 2.2).

2.1.1 Protokoli

Kljub standardizaciji obstaja več različnih protokolov, ki se uporabljajo za komunikacijo z diagnostičnim računalnikom. Tu gre predvsem za razlike na fizičnem (električnem) nivoju. Vsi protokoli so še vedno v uporabi, čeprav se jih nadomešča z novejšim CAN (angl. Controller Area Network) protokolom.

SAE J1850 PWM: [8] Pulse Width Modulation, večinoma v vozilih proizvajalca Ford.

SAE J1850 VWM: [8] Variable Width Modulation, večinoma v vozilih proizvajalca General Motors.

ISO 9141-2: [9] Standard, ki se je uporabljal v večini vozil evropskih in azijskih proizvajalcev, ter v vozilih proizvajalca Chrysler, proizvedenih med 2000 in 2004.

ISO 14230 KWP2000: [10] Keyword Protocol 2000, zelo pogost v vozilih od leta 2003 naprej.

ISO15765-4 (CAN) [11] Controller area network, najnovejši in najbolj izpopolnjen, v uporabi v vseh novejših vozilih (2008-)

2.1.2 Podrobno o izbranem protokolu

SAE J1850 PWM [8] je standardiziran po SAE (Society of Automotive Engineers). Uporablja pulzno-širinsko modulacijo in omogoča prenos podatkov s hitrostjo 41,6 kbit/s [1] Uporabljal se je v vozilih proizvajalca Ford. Uporablja Carrier Sense Multiple Access protokol z dodano Non Destructive Arbitration[2] funkcijo, ki deluje podobno kot Carrier Detection, le da prilagaja hitrost pošiljanja. Uporablja se za izogibanje trkom pri sočasnem dostopu do vodila. Dolžina sporočila + CRC (angl. Cyclic Redundancy Check) je omejena na 12 bajtov. Podatkovni nožici (angl. „pin“) sta druga in deseta. Uporabljata se za prenos podatkov. Stanje visoke napetosti je 5V.

SAE J1850 VWM [8] je zelo podoben 2.1.2 (SAE J1850 PWM) le da omogoča prenos podatkov s hitrostjo 10,4 kbit/s in se uporablja predvsem v vozilih proizvajalca General Motors.

ISO 9141-2 [9] je standardiziran po ISO (International Standards Organisation) . Uporablja asinhrono serijsko komunikacijo in omogoča prenosa podatkov s hitrostjo 10,4 kbit/s. Za komunikacijo se uporablja UART (angl. Universal Asynchronous Receiver/Transmitter). Dolžina sporočila je daljša

[1] kilobitov na sekundo, 1kbit = 1024 bitov. Bit je osnovna enota za prenos podatkov in je lahko 0 ali 1).

[2] `https://www.google.com/patents/US5402420`, 22.10.2016

kot pri SAE protokolih, saj je lahko največ 260 bajtov, od tega je podatkovnih 255 bajtov. Uporablja se v večini vozil evropskih in azijskih proizvajalcev, ter vozilih proizvajalca Chrysler.

ISO 14230 KWP2000 [10] je standardiziran po ISO. Uporablja (dvosmerno) serijsko komunikacijo in je edini od protokolov, ki je kompatibilen s CAN, tako da se (lahko) uporablja tudi v novejših vozilih, ki imajo serijsko vgrajeno CAN vodilo (angl. CAN bus). Tudi tu lahko sporočilo vsebuje do 255 bajtov, hitrost pa lahko niha (odvisno od samega vodila).

ISO15765-4 (CAN) [11] je danes najbolj razširjen protokol, saj ga morajo imeti vsa vozila narejena po letu 2008, ki se prodajajo v ZDA [12]. Standardiziran je po ISO. Hitrost prenosa podatkov je 250 kbit/s, oziroma pri dvojni hitrosti 500 kbit/s. Razvili so ga pri podjetju Robert Bosch GmbH leta 1986 in je edini od protokolov, kjer komponente ECU (senzorji, krmilniki...) ne potrebujejo glavnega računalnika (angl. host computer, tudi master computer), temveč lahko komunicirajo direktno med seboj. To poteka tako, da so vse komponente povezane v (lahko tudi vsaka v svoje) vozlišča (angl. node), ta pa na vodilo in ko eno od njih odda sporočilo, ga prejmejo tudi ostala. Če je sporočilo zanje pomembno ga nadalje obdelajo, sicer ga zavržejo. S tem se bistveno zmanjša število povezav med posameznimi komponentami (topologija vodila namesto „vsak z vsakim“), kar omogoča cenejše in hitrejše vodilo. CAN omogoča prioritete, kar pomeni, da se ob trku pri hkratnem dostopu do vodila sporočilo z višjo prioriteto takoj pošlje še enkrat, medtem ko mora tisto z nižjo prioriteto čakati. Čeprav smo omenili, da je CAN najbolj izpopolnjen, pa še vedno ni dovolj velik poudarek na varnosti, kar je dokazano mogoče izkoristiti za prevzem nadzora nad vozilom (več v poglavju 3). CAN vodil je lahko več, kar vsaj nekoliko oteži napade (bolj pomembne ECU so lahko na ločenem vodilu od manj pomembnih).
Pri testiranju je bilo uporabljeno vozilo s protokolom ISO 9141-2. Ostali protokoli niso bili vključeni (zaradi omejenih finančnih / materialnih sredstev).

2.2 Princip delovanja

Za pridobitev podatkov iz OBD se uporabljajo tako imenovane PID (parameter ID) kode, ki jih definira standard SAE j1979[13], lahko pa tudi sami proizvajalci vozil. Obstaja 10 standardnih načinov delovanja, proizvajalci pa lahko poljubno dodajo še druge. Ti načini delovanja definirajo katere operacije se lahko dogajajo (npr. način delovanja 01 prikaže trenutne podatke), zato imajo iste PID kode drug pomen v različnih načinih delovanja. Samo delovanje je implementirano tako, da ko na vhod pošljemo (veljaven) PID, se ta pošlje po vodilu (angl. vehicle bus), ki povezuje krmilnike. Ko krmilnik prepozna PID, na vodilo spusti željene podatke. Krmilnik ima lahko več PID za katere je odgovoren (npr. krmilnik za gorivo lahko preveri nivo goriva, gostoto goriva, ipd). Z vodila se nato preko adapterja prebere odgovor. Od tu naprej je stvar programa/človeka, da iz odgovora izlušči bistvo. Tega običajno (razen pri razhroščevanju) ne delamo na „roke“, temveč to dela program.

2.3 Android

Za samo testiranje se je uporabil pametni telefon, na katerem teče Android OS, (v času začetka raziskave najnovejša) različica 6.1. Izbran je bil na podlagi sledečih kriterijev:

- Odprtokodnost - v primeru težav pri razvoju je možnost vpogleda v samo kodo sistema in tako poiskati morebitne težave
- Najbolj razširjen - smiseln razvoj testne aplikacije, saj je število potencialnih testerjev večje. Poleg tega je skupnost večja, kar pomeni lažje iskanje pomoči, boljše dokumentirani postopki...
- Cenovno ugodnejše naprave - naprave z Android OS so cenovno precej ugodnejše od konkurenčnih (recimo Apple iPhone).

- Razvoj aplikacij poteka v programskem jeziku Java, ki ga že poznamo, tako da se ni potrebno učiti novih jezikov (recimo Swift, Objective-C, .NET...)

- Razvoj lahko poteka na katerem koli od glavnih operacijskih sistemov (Microsoft Windows, Mac OS, GNU/Linux), kar zmanjša stroške in poveča produktivnost, saj ni potrebe po posebni strojni opremi (Apple) ali na primer tako imenovanem "dual boot" sistemu.

2.3.1 O Androidu

Android je odprtokodni operacijski sistem, ki poganja večino pametnih telefonov (pa tudi tablice, televizije, ure...). Za njim stoji računalniški gigant Google, vodilna sila Android Open Source Project-a, ki razvija Android. Razvoj se je začel leta 2003, prva različica pa je izšla leta 2008. Android temelji na Linux jedru, ki je prosto dostopno. Različica 6.1, poimenovana „Marshmallow", na kateri so se izvajali testi, je bila v času razvoja še najnovejša, danes pa jo je nasledila različica 7, imenovana „Nougat".

2.4 Adapter

Obstaja cel kup adapterjev, ki so se sposobni povezati na OBD na večini vozil. Ti adapterji se v glavnem delijo na dva dela, žične (uporablja se recimo USB povezava) in brezžične (Bluetooth povezava). Glavne razlike:

Primerjava	**Žični adapterji**	**Brezžični adapterji**
Enostavnost	Nerodni (potrebno je imeti ves čas priklopljen kabel	Ni zapletov s kabli, lažja uporaba med vožnjo.
Zanesljivost	Zanesljivi.	Odvisno od povezave in motenj
Cena	Razpon od cenejših do dražjih	Cenovno ugodni

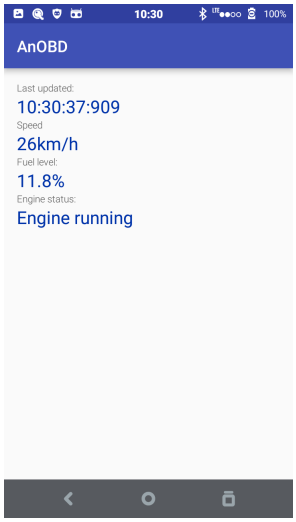
V praksi se izkaže, da večina[3] serviserjev raje uporabi žični adapter, ker je bolj zanesljiv in je z njim manj težav, saj je pri brezžični povezavi potrebno naprave najprej seznaniti (t.i. pairing). Za navadne uporabnike, ki bi želeli med vožnjo opazovati določene podatke, ali pa jih recimo v realnem času obdelati za prikaz zanimivih rezultatov, pa je kabel nepraktičen, poleg tega pa večina mobilnih naprav sploh ne omogoča priklopa le tega.
Obe vrsti adapterjev dvomljive kvalitete lahko kupimo precej ugodno (nekaj evrov za kos). Namenski žični adapterji preverjene kvalitete pa običajno stanejo tja do 200 evrov in več.
Za potrebe testiranja sta se uporabljala Bluetooth adapterja dvomljive kvalitete, ki pa sta se dobro obnesla, kljub nekaterim slabim ocenam na spletu. Adapter deluje le, če je motor vžgan.

2.4.1 Komunikacija

Večina adapterjev podpira vseh pet protokolov iz 2.1.1. To je rešeno z abstrakcijo protokola na višjem nivoju - določen je nabor ukazov, katere mikrokrmilnik pretvori v ukaze po protokolu, ki se dejansko uporablja v vozilu. Med temi mikrokrmilniki je najbolj popularen ELM327 [14] proizvajalca ELM Electronics, ki temelji na starejšem PIC mikrokrmilniku podjetja Microchip Technology.
Na spletu je zaslediti zapise (nepreverjene informacije, [15]), da ELM327 v1.0 ni bil zaščiten pred kopiranjem, tako da so drugi proizvajalci ukradli program in ga uporabili na lastnih mikrokrmilnikih. Ker je (v času nastanka tega besedila) najnovejša različica ELM327 2.2[14], se špekulira, da so imele prejšnje različice hrošče, kar lahko pojasni nedelujoče mikrokrmilnike drugih proizvajalcev (poleg tega, da je bil program napisan za ELM327 mikrokrmilnik).
ELM327 upravljamo preko AT ukazov, ki so običajno sestavljeni iz enega ali več parov, ki pomenijo ukaz. Tako na primer za prepoznavanje protokola

[3]V raziskavi je sodelovalo 5 različnih mehanikov in vsi so se odločili za žični adapter. Rezultati niso nujno odraz dejanskega stanja, a ker to ni bistvenega pomena za raziskavo smo anketirali manjše število.

Slika 2.3: Dva Bluetooth adapterja, ki sta bila uporabljena.



Slika 2.4: Prikaz aplikacije

protokola lahko uporabimo ukaz „AT SP 0“, ki pomeni „Select protocol 0“, oziroma avtomatsko izberi protokol. Po vsakem ukazu prejmemo „OK“, če je bil ukaz razumljiv. Na ta način si olajšamo delo, saj nam ni potrebno skrbeti za to, kateri OBD protokol se uporablja za komunikacijo na nižjem nivoju.

2.5 Testna aplikacija

Za lažje testiranje je bila razvita Android aplikacija. Razvoj je potekal z Android Studiem[4], ki je uradno razvojno okolje za Android aplikacije. Uporablja se programski jezik Java, z dodanim Android ogrodjem (angl. framework). Aplikacija je sestavljena iz enostavnega grafičnega vmesnika (GUI - Graphic User Interface), preko katerega je mogoče izbrati adapter in nato spremljati podatke, ki jih aplikacija prejme iz ECU.

[4] `https://developer.android.com/studio/index.html`, obiskano 14.1.2017

2.5.1 Bluetooth povezava z adapterjem

Aplikacija podpira dva načina povezave, šifrirano (z E0 [16]) in nešifrirano. Za povezavo se uporablja klasična (ne BLE - Bluetooth Low Energy) [17] povezava, pred tem pa je potrebno napravi seznaniti (pairing). Povezava ima domet okrog 10m.

2.5.2 Sporazumevanje z napravo

Po začetni vzpostavljeni povezavi se pošlje nekaj kontrolnih ukazov (echo off, line feed off, timeout), nato pa se določi kateri OBD protokol je v uporabi. V ta namen se uporablja odprtokodna knjižnica **obd-java-api** avtorja Pires [18]. Na ta način si olajšamo nepotrebno delo, saj je knjižnica dovolj dobra.

2.5.3 Branje podatkov s pomočjo aplikacije

Beremo lahko precej različnih podatkov, ki so standardizirani po [13]. Seznam vseh podatkov, ki smo jih prebrali je na voljo v poglavju 4. Katere podatke lahko pridobimo je odvisno tudi od vozila (katere senzorje ima vgrajene), saj nekateri (starejši) avtomobili (preverjeno) ne omogočajo npr. dostopa do količine goriva.
Želeli smo prebrati vsaj stanje odometra, brzinomera in stanje motorja (izklopljen, vklopljen), ker se te stvari v praksi najpogosteje uporabljajo (razne aplikacije, ki nas opomnejo na menjavo olja, izklop motorja po določenem času neuporabe, itd). Na žalost odometer ni del standarda in ga zato ni mogoče enostavno prebrati (razen s posebnimi orodji prilagojenimi za določeno znamko / model, česar pa ne želimo).

2.5.4 Delovanje

Sama aplikacija deluje tako, da se poveže z že poznano napravo (ki jo uporabnik lahko sam izbere) in od nje začne pridobivati vnaprej določene podatke.

2.5.5 Simulacija

Za potrebe razvoja aplikacije smo uporabili orodje OBDSim[19], ki simulira ECU. S tem smo si nekoliko olajšali delo, saj ni potrebno vsake spremembe v aplikaciji testirati na fizičnem vozilu. To prinese za seboj precej bolj produktiven razvoj, saj se ni potrebno ves čas premikati med vozilom in računalnikom, hkrati pa je tudi bolj ekološko, saj ni potrebno ves čas vžigati in izklapljati motorja vozila.

Poglavje 3

Testiranje v praksi

Želeli smo preveriti varnost in stabilnost vozila med uporabo adapterja. To je zelo pomembno, saj kakršnakoli nestabilnost v prometu lahko povzroči velike posledice (prometna nesreča). Raziskovalci so že dokazali, da je možno prevzeti nadzor nad vozilom s pomočjo dostopa do OBD (glej poglavje 3.1), čemur bi se radi na vsak način izognili.

Izvedli smo skupno 4 glavne praktične preizkuse (in več testnih, kjer pa nismo natančno simulirali realnih razmer). V simulaciji je sodelovala oseba s pametnim telefonom in OBD Bluetooth adapterjem, ki je uporabljala testno aplikacijo. Iz varnostnih razlogov ta oseba ni vstopila v promet ampak je zgolj simulirala vožnjo na poligonu.

Raziskali smo tudi bolj zanimive (standardne) ukaze za branje podatkov. Nekatere od teh nam ni uspelo pridobiti, saj jih ECU (starejšega) testnega vozila ni podpiralo (vozilo nima potrebnih senzorjev) - več v poglavju 4.

Ne en ne drugi od obeh testnih adapterjev nam ni uničil OBD oziroma diagnostičnega računalnika. To seveda ne pomeni, da so ti adapterji 100% varni, mogoče smo le naleteli na takšna, ki sta delovala brez težav. Za natančno analizo bi potrebovali precej več teh adapterjev.

3.1 Že znane ranljivosti OBD

V zadnjih letih se odkriva vedno več možnih napadov preko OBD, saj imajo avtomobili vedno več elektronike, ki skrbi za delovanje vozila in udobje potnikov. Vsa ta elektronika pa je prek vodila povezana z diagnostičnim računalnikom (oziroma več njih), kar pomeni, da je preko nje možno dostopati do računalnika. V tem poglavju si bomo pogledali bolj znane napade na OBD, preko katerih lahko vplivamo na delovanje vozila.
Za večino teh napadov je potreben fizični dostop ali pa vsaj neposredna bližina vozila. Te napade bi bilo možno izvesti tudi oddaljeno, če bi bilo vozilo povezano v internetno omrežje (glej poglavje 3.2).

DoS (Denial of Service) napad [20] Najenostavnejši napad. Vse kar je potrebno narediti je, da na vodilo pošiljamo dovolj veliko število (točno število ni natančno definirano, saj je odvisno od konkretnega primera) zahtev ali odgovorov nanje. Pri CAN protokolu moramo paziti še na to, da nastavimo prioriteto paketa na najvišjo, sicer nas lahko kdo (recimo krmilnik z višjo prioriteto) prekine in tako zaustavi napad. Cilj tega napada je onemogočanje komunikacije med napravami na vodilu. Na ta način lahko povzročimo težave, saj so nekatere naprave medsebojno odvisne (recimo nadzor porabe goriva deluje le, ko je motor vžgan).

Ponareditev podatkov [20] Če nam uspe prevzeti nadzor nad eno od naprav na vodilu, lahko s pomočjo te naprave na vodilo pošiljamo poljubne podatke. Cilj tega napada je, da ponaredimo podatke, ki si jih naprave pošiljajo med seboj. Na ta način lahko vplivamo na akcije, ki jih glede na prejete podatke izvede določena naprava, pa tudi na akcije samega voznika (če mu na primer prikažemo, da vozi z nižjo hitrostjo kot v resnici, potem bo zelo verjetno vozil prehitro).
Ponarejanje podatkov je precej težje izvedljiv napad, saj je potrebno poznati strukturo sporočila na vodilu, ki je običajno odvisna od modela in proizvajalca vozila. Poleg tega so naprave v vozilu lahko omejene tudi na

določeno podomrežje znotraj avtomobilskega omrežja in tako ne morejo poslati sporočila do vseh naprav.
S pomočjo zgoraj omenjenih napadov lahko izvedemo večino drugih napadov, ki so zelo podobni opisanima, le da uporabimo druge vstopne točke (npr. namesto vstopne točke OBD lahko uporabimo recimo avtoradio).

3.2 Napadi preko mreže

V zadnjih letih smo priča porastu IoT (angl. Internet of Things). Tudi vozila sledijo temu trendu in danes imajo novejša vozila možnost povezljivosti z Bluetooth, WiFi, pa tudi z drugimi vozili in okolico [21]. Nekateri temu pravijo kar IoV (angl. Internet of Vehicles). Vse to se seveda sliši zelo lepo, a proizvajalci vozil zelo redko pomislijo na varnost pred vdori v tak sistem. Po našem mnenju bodo v nekaj letih precej bolj popularni napadi preko mreže, saj ni potrebno biti v bližini vozila. Pogledali si bomo nekaj napadov, ki so jih izvedli razni strokovnjaki, pa tudi kakšno je trenutno stanje in kakšna so pričakovanja za prihodnja leta.

Pregled Po ocenah Garnter-ja bo do leta 2020 na svetu približno 250 milijonov povezanih vozil (angl. Connected vehicle) [22]. Ta vozila si bodo lahko med seboj izmenjevala informacije (recimo o zastojih). To pomeni, da bo možnosti za vdor ogromno. Že danes imamo poročila o prevzemu nadzora nad različnimi deli avtomobila, kot so recimo izklop zavor, vklop klimatske naprave, zaklepanje vrat, nadzor nad volanom... [23], [24], [25], ki jih proizvajalci vozil sploh ne morejo zaznati [26].
Pojavi se vprašanje, ali so starejša vozila, oziroma vozila nižjega cenovnega razreda z manj elektronike zaradi tega bolj varna? Izkaže se da, da je v omrežje takšnih vozil lažje vdreti, a težko povzročiti posledice, saj se prek vodila ne pošiljajo pomembni podatki (za to se uporabljajo ločeni vodniki) [25].
Poleg tega se v zadnjih letih v avtomobile vgrajuje precej zabavne elektro-

nike, ki nam omogoča npr. pošiljanje SMS, brskanje po Internetu ali uporabo mobilnih aplikacij kar na vgrajeni avtomobilski opremi (zvočniki, zaslon na dotik, mikrofon), pa tudi nadzor nad vozilom s pomočjo npr. pametne ure [27]. Da nam nekdo lahko ukrade uro, z njeno pomočjo poišče vozilo, ga odklene in odpelje najbrž ni potrebno posebej omenjati.

Zasebnost Velika večina povezanih vozil pošilja povratne informacije k proizvajalcu vozil [28], ti pa lahko s temi podatki počnejo kar želijo. Uradno se ti podatki uporabljajo za izboljšanje uporabniške izkušnje, a smo lahko upravičeno zaskrbljeni, saj so v vozilih nameščene kamere, mikrofoni, GNSS (angl. Global Navigation Satellite System) moduli..., ki podatke pošiljajo na strežnike izven našega nadzora. V kolikor jih želimo odstraniti pa tvegamo izgubo garancije [29]. To področje spremlja tudi združenje avtomobilskih proizvajalcev - Auto Alliance [30], ki se trudi, da bi vozila ustrezala vsaj minimalnim standardom zasebnosti.

Samovozeča vozila Zadnja leta so zelo popularna tema tudi samovozeča ali avtonomna vozila. Veliko proizvajalcev vozil, recimo Tesla, BMW, General Motors, Toyota, pa tudi druga podjetja, recimo Google, Apple, Nvidia, Uber... preizkušajo samovozeča vozila. Ta imajo že zaradi svoje narave še veliko večjo povezljivost, kar pomeni še dodatne možnosti vdora v primerjavi z navadnimi vozili. Večina teh vozil uporablja senzor, imenovan Lidar, s pomočjo katerega zaznava ovire v okolici vozila. Raziskovalcem je uspelo pretentati Lidar s pomočjo laserja [31]. Na ta način lahko vozilo poljubno usmerjajo ali ustavijo.

Pričakovanja v prihodnosti Pričakuje se, da bo samovozečih vozil v nekaj letih precej več (po navedbah Business Insider kar deset milijonov [32]), zato smo lahko upravičeno zaskrbljeni nad varnostjo v prometu. Na predlog podjetja Intel Security se je septembra 2015 ustanovila neprofitna organizacija ASRB (angl. Automotive Security Review Board), kasneje preimenovana v FASTR (angl. Future of Automotive Security Technology Research),

ki spremlja razmere na področju varnosti v avtomobilski industriji. Med bolj znanimi člani je poleg podjetja Intel tudi Uber. Cilji organizacije so pripraviti vozila na ta način, da ima voznik / lastnik nadzor nad zasebnostjo in popolnim delovanjem vozila. Zelo veliko je odvisno tudi od kupcev, saj veliki večini za zasebnost ni mar (kar opazimo tudi na drugih trgih, recimo pri nakupu mobilnih naprav).

Poglavje 4

Pregled rezultatov

V tem poglavju si bomo pregledali rezultate branja podatkov iz OBD s pomočjo Bluetooth adapterja.

4.1 Branje podatkov

Branje podatkov smo izvedli tako, da smo adapter priklopili v standardni OBD vhod na vozilu, se nanj povezali s telefonom, nato pa smo se z vozilom vozili po poligonu. S pomočjo prej napisane testne aplikacije so se v tem času na telefonu prikazovali podatki, ki smo jih sproti preverjali na števcih v vozilu. S pomočjo adapterja nam je uspelo prebrati vse podatke, ki jih ECU podpira. Podatke smo pridobili s pomočjo ELM327 ukazov. Gre za starejše vozilo (letnik 2002), ki uporablja protokol ISO 9141-2:. Nekatere podatke (npr. podatek iz senzorja kisika v motorju), ki se nam niso zdeli uporabni smo izpustili. Podatki označeni z * niso bili uspešno prebrani (zaradi odsotnosti senzorjev/krmilnikov). Pridobili smo podatke o:

podatek	**način delovanja**	**PID**
Številu obratov motorja	01	0C
Hitrosti vozila	01	0D
Položaju pedala za plin (pedala za gas)	01	11
Nivoju goriva (v %)	01	2F
Napetosti v ECU (običajno skoraj enaka akumulatorju)	01	42
Temperaturi okolja*	01	46
Tipu goriva*	01	52
Ponastavitev oznak za napake	04	/
VIN (ID vozila)	09	02

4.2 Stabilnost

Na testnem vozilu smo uporabljali oba adapterja dvomljivega slovesa, a vozilo še vedno deluje brez težav. Imeli smo sicer nekaj težav s povezljivostjo med adapterjem in mobilno napravo, ki smo jih običajno razrešili tako, da smo adapter (večkrat) iztaknili in ponovno priklopili. Nato smo se nanj zopet uspeli povezati. Adapter ne deluje, če je motor izklopljen. Drugih težav nismo opazili. Pravilnost podatkov smo preverili s primerjavo RPM, hitrosti vozila in nivoja goriva glede na števec v vozilu.

Poglavje 5

Zaključek

Ugotovili smo, da se da s pomočjo adapterja iz OBD prebrati podatke, ki jih lahko uporabimo na različne načine, recimo v mobilnih aplikacijah ki nas spomnejo na servisiranje vozila, beležijo poti (v kombinaciji z GNSS v mobilni napravi), ocenijo vožnjo (avtomobilsko zavarovanje po meri), itd. S pomočjo cenovno ugodnega adapterja lahko torej naredimo kakšen zanimiv projekt (nekaj idej lahko najdemo tudi v [33]) in še vedno ostanemo znotraj proračuna povprečnega študenta. Poleg mobilnih aplikacij bi lahko tudi sprogramirali tudi kakšen mikrokrmilnik, nadomestili nedelujoče števce...
Raziskali smo tudi področje povezanih in samovozečih vozil, za katero pričakujemo, da bo vedno bolj popularno. Še vedno pa ostajajo vprašanja varnosti (pred zunanjimi napadi) in zasebnosti (vedno več podatkov iz avtomobila se shranjuje na strežnike nad katerimi nimamo nadzora). Smo torej v času velikih sprememb na področju avtomobilizma, kam se bo vse skupaj obrnilo pa bomo videli v prihodnjih letih.

Literatura

[1] USA. Vehicle service information 1995 final rule. https://www.gpo.gov/fdsys/pkg/FR-1995-08-09/pdf/95-18867.pdf, 1995. Zadnji dostop: 13.01.2017.

[2] European Union. Directive 98/69/ec of the european parliament and of the council. http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CONSLEG:1998L0069:19981228:EN:PDF, 1998. Zadnji dostop: 13.01.2017.

[3] Zavarovalnica Triglav. Drajv. http://www.triglav.si/drajv. Zadnji dostop: 13.01.2017.

[4] Arijit Chowdhury, Tapas Chakravarty, and P Balamuralidhar. Estimating true speed of moving vehicle using smartphone-based gps measurement. In *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3348–3353. IEEE, 2014.

[5] Many Authors. Attention! "bad"bluetooth adapters! http://www.incardoc.com/forum/default.aspx?g=posts&t=84, 2014. Zadnji dostop: 13.01.2017.

[6] SAE. Sae j1962. http://standards.sae.org/j1962_201207/. Zadnji dostop: 13.01.2017.

[7] ISO. Iso 15031-3. http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=29021, 2004. Zadnji dostop: 13.01.2017.

[8] SAE. Sae j1850. http://standards.sae.org/j1850_201510/, 2015. Zadnji dostop: 13.01.2017.

[9] ISO. 9141-2. https://www.iso.org/obp/ui/#iso:std:iso:9141:-2:en, 2015. Zadnji dostop: 13.01.2017.

[10] ISO. Kwp2000. http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=69115, 2016. Zadnji dostop: 13.01.2017.

[11] ISO. Can. http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=63648, 2015. Zadnji dostop: 13.01.2017.

[12] US Goverment. Code of federal regulations. https://www.gpo.gov/fdsys/pkg/CFR-2013-title40-vol20/xml/CFR-2013-title40-vol20-sec86-1806-05.xml, 2013.

[13] SAE. Sae j1979 standard. http://en.sae.org/standards/groundvehicle/j1979_201009/. Zadnji dostop 30.01.2017.

[14] ELM Electronics. Elm327. https://www.elmelectronics.com/products/ics/obd/, 2016. Zadnji dostop: 13.01.2017.

[15] CAN in Automation (CiA). Can newsletter. https://can-newsletter.org/tools/tools-miscellaneous/nr_elm327_wgsoft_140123/. Zadnji dostop: 13.01.2017.

[16] Juha T Vainio. Bluetooth security. In *Proceedings of Helsinki University of Technology, Telecommunications Software and Multimedia Laboratory, Seminar on Internetworking: Ad Hoc Networking, Spring*, 2000.

[17] Inc Bluetooth SIG. Bluetooth. https://www.bluetooth.com/specifications/bluetooth-core-specification, 2017. Zadnji dostop: 13.01.2017.

[18] Paulo Pires. obd-java-api. https://github.com/pires/obd-java-api. Zadnji dostop: 14.01.2017.

[19] Gary 'Chunky Ks' Briggs. Obdsim. https://icculus.org/obdgpslogger/obdsim.html. Zadnji dostop: 01.11.2016.

[20] Eric Evenchick. Hopping on the can bus. https://www.blackhat.com/docs/asia-15/materials/asia-15-Evenchick-Hopping-On-The-Can-Bus.pdf. Predavano na konferenci Black Hat Asia 2015. Zadnji dostop 13.2.2017.

[21] Uichin Lee and Mario Gerla. A survey of urban vehicular sensing platforms. *Computer Networks*, 54(4):527–544, 2010.

[22] Gartner. Predicts 2015: The internet of things. https://www.gartner.com/newsroom/id/2970017. Zadnji dostop: 23.02.2017.

[23] Andy Greenberg. Hackers remotely kill a jeep on the highway—with me in it. *Wired*, 7:21, 2015.

[24] Karl Koscher, Alexei Czeskis, Franziska Roesner, Shwetak Patel, Tadayoshi Kohno, Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, and Stefan Savage. Experimental security analysis of a modern automobile. http://www.autosec.org/pubs/cars-oakland2010.pdf. Zadnji dostop: 13.01.2017.

[25] Yu Zhang, Binbin Ge, Xiang Li, Bin Shi, and Bo Li. Controlling a car through obd injection. In *Cyber Security and Cloud Computing (CSCloud), 2016 IEEE 3rd International Conference on*, pages 26–29. IEEE, 2016.

[26] Ed Markey. Tracking & hacking. http://www.markey.senate.gov/imo/media/doc/2015-02-06_MarkeyReport-Tracking_Hacking_CarSecurity%202.pdf. Zadnji dostop: 23.02.2017.

[27] Volvo. Control your volvo from watch. https://www.media.volvocars.com/us/en-us/media/pressreleases/163015/control-your-volvo-car-from-your-apple-watch. Zadnji dostop: 23.02.2017.

[28] Maurice Schellekens. Car hacking: navigating the regulatory landscape. *Computer Law & Security Review*, 32(2):307–315, 2016.

[29] Erin Biba. Your car is not your friend. http://www.bbc.com/autos/story/20160809-your-car-is-not-your-friend. Zadnji dostop: 23.02.2017.

[30] Auto Alliance. Privacz. https://autoalliance.org/connected-cars/automotive-privacy-2. Zadnji dostop: 23.02.2017.

[31] Jonathan Petit, Bas Stottelaar, Michael Feiri, and Frank Kargl. Self driving and connected cars fooling sensors and tracking drivers. https://www.blackhat.com/docs/eu-15/materials/eu-15-Petit-Self-Driving-And-Connected-Cars-Fooling-Sensors-And-Tracking-Drivers.pdf. Zadnji dostop: 23.02.2017.

[32] John Greenough. 10 million self-driving cars will be on the road by 2020. *Business Insider*, 2016.

[33] Malintha Amarasinghe, Sasikala Kottegoda, Asiri Liyana Arachchi, Shashika Muramudalige, HMN Dilum Bandara, and Afkham Azeez. Cloud-based driver monitoring and vehicle diagnostic with obd2 telematics. In *Advances in ICT for Emerging Regions (ICTer), 2015 Fifteenth International Conference on*, pages 243–249. IEEE, 2015.